Express Mail Label No. EL915360906US

PROTECTING DATA IN A NETWORK ATTACHED STORAGE DEVICE

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] Not applicable.

STATEMENT REGARDING FEDERALLY SPONSORED RESEARCH OR DEVELOPMENT

[0002] Not applicable.

BACKGROUND OF THE INVENTION

Field of the Invention

[0003] The present invention relates generally to security on a computer network. More particularly, the invention relates to protecting data stored on a network attached storage device. Still more particularly, the invention relates to storing data in encrypted form on a network attached storage device and reducing performance impact.

Background of the Invention

[0004] Security is a concern for many computer systems, particularly those computer systems that contain sensitive information. In some applications, a storage device is coupled to a network and accessible by various computers also coupled to the network. Such storage devices are referred to as network attached storage ("NAS") devices. A security issue arises in the context of a network to which unrelated entities have access. If such a network includes a NAS device to which each entity can access, a security system should be implemented to prevent one entity from accessing the data stored on the NAS by an unrelated entity.

[0005] In one type conventional security systems, each entity wishing to store data on the NAS encrypts the data and transmits the encrypted data to the NAS device. Upon receipt of the encrypted data, the NAS device decrypts the data and stores the decrypted data on the device. This security system minimizes the risk that an unauthorized entity can intercept a transmission and recover the data in a useful form. Because the transmission includes encrypted data, the unauthorized entity will find the data unless it knows or figures out how to decrypt the message.

[0006] Although generally satisfactory, this approach is not without its shortcomings and limitations. For instance, once the NAS successfully decrypts the data and stores it therein, the unencrypted data can be accessed by unauthorized entities.

[0007] Further still, encryption typically involves a pair of "keys." The data may be encrypted with a "public" key by the entity transmitting the data and then decrypted by the NAS device using a related "private" key. The public-private key pair is unique to each entity. That is, each entity has a public-private key pair that is different from the key pairs of the other entities. As its name implies, the private key is highly confidential and protecting the security of the private key itself is of paramount concern. If the private keys were stored on the NAS, a security problem would arise if unauthorized entities were to obtain the private keys. With the private keys in the hands of an unauthorized entity, any confidential data transmitted to the NAS may be compromised. Various security protocols have been suggested and implemented to deal with this concern, but no security system is 100 % fool proof.

[0008] Another shortcoming is that the NAS device must incur the task of decrypting the incoming data to extract the original unencrypted data. This task takes time and processing power that perhaps could be used to do other tasks. At a minimum, a NAS that does not have to perform the decryption task would be faster and thus less expensive. Accordingly, a security mechanism is needed which addresses these issues.

BRIEF SUMMARY OF THE INVENTION

[0009] The problems noted above are solved in large part by a computer system comprising at least one computer and at least one storage device coupled together via a network. The computers can store data on and read data from the storage devices. Preferably, the computers transmit data and encrypt the payload as part of the transmission process. This entire packet is transmitted to the storage device where the packet is received, and the encrypted payload is stored still in encrypted form. When a computer requests data that is stored on the storage device, the storage device retrieves the requested data (which is encrypted) and transmits the still encrypted data to the computer that requested the data. The requesting computer then decrypts the encrypted data and recovers the original data.

[0010] In an alternative embodiment, the storage device again encrypts the already encrypted data when sending the data back to the computer. The twice encrypted data is then received by the requesting computer and twice decrypted to recover the original data. Further still, digital signatures can be implemented to help verify the origin, authenticity, and integrity of the data.

[0011] By storing encrypted data on the storage device, without first decrypting it, no encryption/decryption keys need be stored on the storage device. Accordingly, security is increased by not having the data stored in an unprotected manner on the storage device. Further, the storage device need not incur the resource overhead associated with decrypting data. These and other advantages will become apparent upon reviewing the following disclosure.

BRIEF DESCRIPTION OF THE DRAWINGS

[0012] For a detailed description of the preferred embodiments of the invention, reference will now be made to the accompanying drawings in which:

[0013] Figure 1 shows a block diagram of a computer system including computers and a network attached storage device coupled together via a network;

[0014] Figure 2 shows an exemplary data packet format used to transmit data packets across the network;

[0015] Figure 3 shows one preferred embodiment for transmitting encrypted data from a computer to the storage device where the data is stored in its encrypted form; and

[0016] Figure 4 shows an alternative embodiment in which data is twice encrypted when being sent from the storage device to the computer requesting the data.

NOTATION AND NOMENCLATURE

[0017] Certain terms are used throughout the following description and claims to refer to particular system components. As one skilled in the art will appreciate, computer companies may refer to a component and sub-components by different names. This document does not intend to distinguish between components that differ in name but not function. In the following discussion and in the claims, the terms "including" and "comprising" are used in an open-ended fashion, and thus should be interpreted to mean "including, but not limited to…". Also, the term "couple" or "couples" is intended to mean either a direct or indirect electrical connection. Thus, if a first device couples to a second device, that connection may be through a direct electrical connection, or through an indirect electrical connection via other devices and connections. To the extent that any term is not specially defined in this specification, the intent is that the term is to be given its plain and ordinary meaning.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0018] In general, the preferred embodiments of the invention described below involve transmitting encrypted data to a network attached storage ("NAS") device and storing the data in the NAS device in encrypted form, rather than decrypting it before storage therein. The following embodiments describe several variations on this theme.

[0019] Referring first to Figure 1, a computer system 90 is shown as comprising one or more computers 102 and a NAS 104 coupled together via a network link 100. The system 90 may comprise a local area network ("LAN"), a wide area network ("WAN"), such as the Internet, and, in general, include any type of communication infrastructure through which computers 102 and NAS 104 can communicate with one another. Preferably, each computer 102 can write data to and/or read data from NAS 104 over the network link 100. The computers 102 can be any suitable type of computer, workstation, mainframe, or, in general any entity that can access a storage device in a network. The NAS 104 is any suitable type of mass storage device such as a hard disk drive, R/W CD ROM, tape drive, etc, and thus includes some form of a non-volatile storage medium on which data can be stored. The NAS 104 includes logic (not shown), which may be implemented in a network interface card ("NIC") logic or in software executed by a processor contained in the NAS that performs the functions described herein. The functions described below attributable to the computers 102 also may be implemented in a NIC (not specifically shown) that preferably is included in each computer for communicating over the network link 100. However, one of ordinary skill in the art will understand that there are many ways to implement the functionality described herein (*e.g.*, hardware, software, a combination of hardware and software) and the claims which follow should not be limited to any particular implementation.

[0020] Data is transmitted over the network link 100 preferably in the form of packets such as that shown in Figure 2. As shown, packet 110 includes a header portion 112, a footer portion 114, and a data payload 116. As is well known in the art, the header contains information (*e.g.*, IP address, routing information, etc.) that permits the network 100 to determine how to route the packet from the source to the destination. The footer contains information that indicates the end of the packet. The header and/or footer may also contain cryptographic integrity/authenticity metrics (ala a digital signature) and are used to validate the integrity/authenticity of the data prior to storing the encrypted data on the storage device. These metrics preferably are secure hashes and digital signatures. The data payload 116 contains the data, which may include, data, commands or any type of information, to be transmitted between computers 102 and NAS 104.

[0021] In accordance with the preferred embodiment, when a computer 102 uses the network link 100 to transmit to the NAS 104 a packet 110 containing a data payload 116, the data payload preferably is encrypted. Any suitable encryption algorithm now known or later developed can be used such as "DES", "AES", "Blowfish," and the like. In addition any suitable networking protocol now known or later developed can be used such as "IPSEC" or "SSL." While the specific examples given in this disclosure are of the current commonly used asymmetric cipher or public-key/private-key algorithm type, nothing precludes the embodiment being realized using a symmetric cipher or secret-key algorithm. The data, in encrypted form, is stored in the NAS's non-volatile memory. In contrast to conventional storage techniques, the data is not decrypted before being stored on the NAS.

[0022] Figures 3, and 4 illustrate variations on this preferred technique in which encrypted data is stored on the NAS 104, rather than unencrypted data. Figures 3 and 4 illustrate the process flow for how data is encrypted by a computer, transmitted to a NAS, stored on the NAS and how NAS data is retrieved and provided to the computer. Each figure shows two communication paths—A and B. The A path in each figure shows the process for sending data from a computer to the NAS 104 for storage therein, while the B path shows the process for retrieving data from storage in the NAS and transmitting it to the computer.

[0023] Referring first to Figure 3, a data file 120 (which may also be a data stream, a block of data or other type of data unit) is turned into a data packet 128 by steps 122. As such, a header 132 and a footer 134 are created. The payload is encrypted preferably using the user's public key (although a secret key can also be used) to form an encrypted data payload 130 and the header 134, encrypted data payload 130 and footer 136 are assembled together into a packet 128 as noted above with regard to Figure 2. The key used to encrypt the file 120 may be stored in the computer or otherwise accessible to it.

[0024] That packet 128 containing encrypted data is transferred across network link 100 to NAS 104 where the header and footer are stripped off and the encrypted data payload is obtained and stored as encrypted data 140 on NAS 104. Data that is stored on NAS 104 in encrypted form obviously eliminates the NAS 104 from having to decrypt the data as is required in some conventional systems. Thus, no decryption keys are necessary and no keys need be stored on NAS 104.

[0025] In the B path, in which data flows from the NAS to a computer requesting the data, an encrypted data file 148 is turned into a packet 152 (steps 150) by NAS 104. In accordance with these steps, a header 154 and a footer 158 are created to permit the network link 100 to route the packet to a destination computer 102. The already encrypted data file 148, which is retrieved from non-volatile memory in the NAS, is included in the packet 152 as encrypted data payload 156 as shown. The packet 152 is then transmitted across the network link 100 to the destination computer where steps 160 are applied by the computer to strip off the header and footer to recover the encrypted data file. The encrypted file is then decrypted by the computer 102 in step 162 using a private key (or public key if a private key was used to encrypt the data initially) to transform the data into its unencrypted format. Thus, both the encryption and decryption processes are performed by the source of the data (*i.e.*, the computers 102), not the NAS 104, and, accordingly, both the public and private keys used in the encryption/decryption process are stored on, or are accessible to, the computer 102.

[0026] An alternative embodiment is shown in Figure 4. The process in path A for encrypting the data file, creating the data packet, transmitting the packet across the network, retrieving the encrypted data payload in the packet and storing the data in encrypted form on the NAS is the same as described above with regard to Figure 3. The difference in Figure 4 pertains to path B when a computer 102 accesses encrypted data from NAS 104. In that regard, the encrypted file 148 to be transmitted to the requesting computer 102 is processed by steps 180 by which a packet 184 is created. The packet 184 includes a header 154 and footer 158 as before, but the encrypted data file 148 is encrypted again (this time by the NAS) to produce a "supra-encrypted" data payload 182 (*i.e.*, twice encrypted data). The packet 184 then is transferred from the NAS 104 to the destination computer 102. In steps 186, the computer 102 strips off the head and footer, decrypts the supra-encrypted data payload to recover the originally encrypted file 148. The encrypted file 148 is then decrypted again in 188 to recover the original unencrypted data file 190.

Thus, in the embodiment of Figure 4, the computer twice decrypts the data received from the NAS 104.

[0027] In the embodiment of Figure 4, the encrypted file 148 can be supra-encrypted using a public key associated with the destination computer or the entity or person owning or operating the computer. The private key necessary to decrypt the supra-encrypted data payload in step 186 is stored on or is accessible to the computer 102. As such, one key is stored on, or accessible to, the NAS 104 and the corresponding other key is stored on, or accessible to, the computer 102. Requiring a private key to decrypt the supra-encrypted data advantageously makes it difficult, if not impossible, for an unauthorized person (not having the private key) to intercept and access the data. The public/private keys used to encrypt the file 120 and decrypt the decrypted supra-encrypted file in step 188 preferably are both stored on or accessible to the computer 102 and preferably are different than the keys used to create the supra-encrypted data payload in 180 and decrypt the supra-encrypted data in 186 (although they can be the same if desired).

[0028] In addition, a digital signature can be applied to the packets as they are transmitted from the computer across the network 100 to the NAS 104. The digital signature, which can be applied in accordance with any well-known or later developed techniques, are then used by NAS 104 to verify the authenticity of the packet (*i.e.*, that the packet indeed did originate from a certain computer 102).

[0029] One last embodiment would handle the case in which the networking protocol uses a predetermined or dynamically generated session key. If dynamically generated, the session key can be negotiated in any suitable manner between the computer and NAS. In this case the session key (K_s) could be stored on the requestor's machine and associated with the file being sent to the NAS. When the requestor asked for the file back, the key (K_s) could be looked up in the requestor's database. This key would then be used to decrypt the file. The decryption could take place either after it was transferred, or during the transferal.

[0030] The above discussion is meant to be illustrative of the principles and various embodiments of the present invention. Numerous variations and modifications will become apparent to those skilled in the art once the above disclosure is fully appreciated. For example, although the embodiments described above have been presented in the context of a network attached storage device coupled to a computer network, in general, the principles apply to the transfer from one point to another of any type of data across any type of network. It is intended that the following claims be interpreted to embrace all such variations and modifications.